



Article

Assessment of Automated Flow Cytometry Data Analysis Tools within Cell and Gene Therapy Manufacturing

Melissa Cheung ^{1,*}, Jonathan J. Campbell ², Robert J. Thomas ¹, Julian Braybrook ² and Jon Petzing ¹

¹ Centre for Biological Engineering, Loughborough University, Loughborough LE11 3TU, Leicestershire, UK; r.j.thomas@lboro.ac.uk (R.J.T.); j.petzing@lboro.ac.uk (J.P.)

² National Measurement Laboratory, LGC, Queens Road, Teddington TW11 0LY, Middlesex, UK; jonathan.campbell@lgcgroup.com (J.J.C.); julian.braybrook@lgcgroup.com (J.B.)

* Correspondence: m.cheung@lboro.ac.uk

Abstract: Flow cytometry is widely used within the manufacturing of cell and gene therapies to measure and characterise cells. Conventional manual data analysis relies heavily on operator judgement, presenting a major source of variation that can adversely impact the quality and predictive potential of therapies given to patients. Computational tools have the capacity to minimise operator variation and bias in flow cytometry data analysis; however, in many cases, confidence in these technologies has yet to be fully established mirrored by aspects of regulatory concern. Here, we employed synthetic flow cytometry datasets containing controlled population characteristics of separation, and normal/skew distributions to investigate the accuracy and reproducibility of six cell population identification tools, each of which implement different unsupervised clustering algorithms: Flock2, flowMeans, FlowSOM, PhenoGraph, SPADE3 and SWIFT (density-based, *k*-means, self-organising map, *k*-nearest neighbour, deterministic *k*-means, and model-based clustering, respectively). We found that outputs from software analysing the same reference synthetic dataset vary considerably and accuracy deteriorates as the cluster separation index falls below zero. Consequently, as clusters begin to merge, the flowMeans and Flock2 software platforms struggle to identify target clusters more than other platforms. Moreover, the presence of skewed cell populations resulted in poor performance from SWIFT, though FlowSOM, PhenoGraph and SPADE3 were relatively unaffected in comparison. These findings illustrate how novel flow cytometry synthetic datasets can be utilised to validate a range of automated cell identification methods, leading to enhanced confidence in the data quality of automated cell characterisations and enumerations.

Keywords: flow cytometry; automated data analysis tools; ATMP manufacturing; regulatory compliance



Citation: Cheung, M.; Campbell, J.J.; Thomas, R.J.; Braybrook, J.; Petzing, J. Assessment of Automated Flow Cytometry Data Analysis Tools within Cell and Gene Therapy Manufacturing. *Int. J. Mol. Sci.* **2022**, *23*, 3224. <https://doi.org/10.3390/ijms23063224>

Academic Editor: Stefano Papa, Paula Fernandez and Claudio Ortolani

Received: 27 January 2022

Accepted: 11 March 2022

Published: 17 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Flow cytometry is a single-cell analytical technique widely applied within manufacturing of advanced therapy medicinal products (ATMPs) and tissue engineered products to measure cell product characteristics, in accordance with Good Manufacturing Practice (GMP) and quality guidelines laid out by regulatory authorities such as the European Medicines Agency (EMA) and the US Food and Drug Administration (FDA) [1,2].

Typical ATMP drug product critical quality attributes (CQAs) evaluated by flow cytometry include identity, purity, potency, quantity, and viability [3]. These CQAs are usually measured from initial sample reception, and at each substantial manipulation step until final characterisation for product release.

Since flow cytometry plays a critical role in ATMP manufacture, the need for continual development of best practice, along with standardisation within the field is well recognised [4]. For instance, the British Pharmacopoeia recently prioritised the preparation of authoritative guidance on the application of flow cytometry for its cell and gene therapy stakeholder communities [5]. Such guidance documents cover the major sources of variation in flow cytometry, these being: starting materials and reagents, equipment, sample

preparation, and data analysis. The lattermost factor, data analysis (manual gating), is a significant source of variation, and its removal from the analytical process has been shown to reduce inter-laboratory variation by as much as 5–20% [6]. Manual setting of gates is difficult to reproduce, subjective [7,8], and potentially biased in favour of a value that aids product release and a successful manufacturing run.

The data analysis aspect of flow cytometry has rapidly evolved in recent years with the development of a wide array of automated cell population identification software that implement unsupervised and supervised machine learning algorithms [9]. These include tools for dimensionality reduction (e.g., t-SNE and UMAP), clustering (e.g., FlowSOM, SPADE and SWIFT), scaffold maps, trajectory inference, and classification and regression (reviewed in [10]).

Previous work from benchmarking studies have suggested certain automated methods were able to reliably replicate manual gating. For example, the FlowCAP competitions saw several algorithms (ADICyt, SamSPECTRAL, and flowMeans) score highly on accuracy in cell identification challenges [11]. Similarly, a comparison of clustering methods focussed on high-dimensional data identified FlowSOM, X-shift, PhenoGraph amongst others as high performing [12]. However, datasets used for these critical assessments often come from a limited range of cell or disease models. Efforts from the recently launched National Institute of Standards and Technology (NIST) Flow Cytometry Standards Consortium to develop biological reference materials, reference data and reference methods are not yet applicable to address the sources of variability from automated data analysis software [13]. Although uptake of these advanced automated data analysis tools within ATMP manufacturing is largely unknown, a recent survey of the clinical community suggests that 20% of clinical laboratories sometimes or usually use them [9], and this number is expected to increase as the toolsets available mature.

Related machine learning-based computational technologies intended for patient diagnostic, treatment or preventative purposes are authorised by the FDA under Software as a Medical Device (SaMD) regulations [14], with a further proposed Artificial Intelligence and Machine Learning (AI/ML)-based SaMD regulatory framework [15]. To date, it appears that no such software focussed on analysis of flow cytometry data have been approved [16]. Relevant ISO/IEC standards for SaMDs include IEC 62304:2006, which defines the life cycle requirements for medical devices software to ensure safety and effectiveness, and ISO 14971:2019, which establishes the application of risk management to medical devices [17,18]. New guidance and efforts to standardise AI/ML in health care are beginning to emerge [19,20], but there is a lack of specific regulatory guidance on their use in the manufacturing of medicines.

The biomanufacturing community faces a challenge on how to compliantly adopt automated flow cytometry data analysis software in ATMP biomanufacturing process controls. While these automated analysis tools have the potential benefit to improve the quality of ATMPs and the capacity to minimise operator variation, in many cases, confidence in these nascent technologies has yet to be fully established. Specifically, a gap currently exists in the toolsets available for standardisation and testing of automated flow cytometry data analysis methods, potentially leaving manufacturers unable to demonstrate a method is fit for its intended purpose, and limiting the trust and transparency of these software tools among users and regulators.

In order to address these issues, synthetic flow cytometry datasets have been previously developed to aid validation of automated cell population identification tools and were shown to successfully mimic real cell data [21]. These synthetic datasets demonstrated clear similarities in cell distribution characteristics when compared against real-world flow cytometry data (Figure 1), and therefore can be used as credible substitutes to represent actual experimental data.

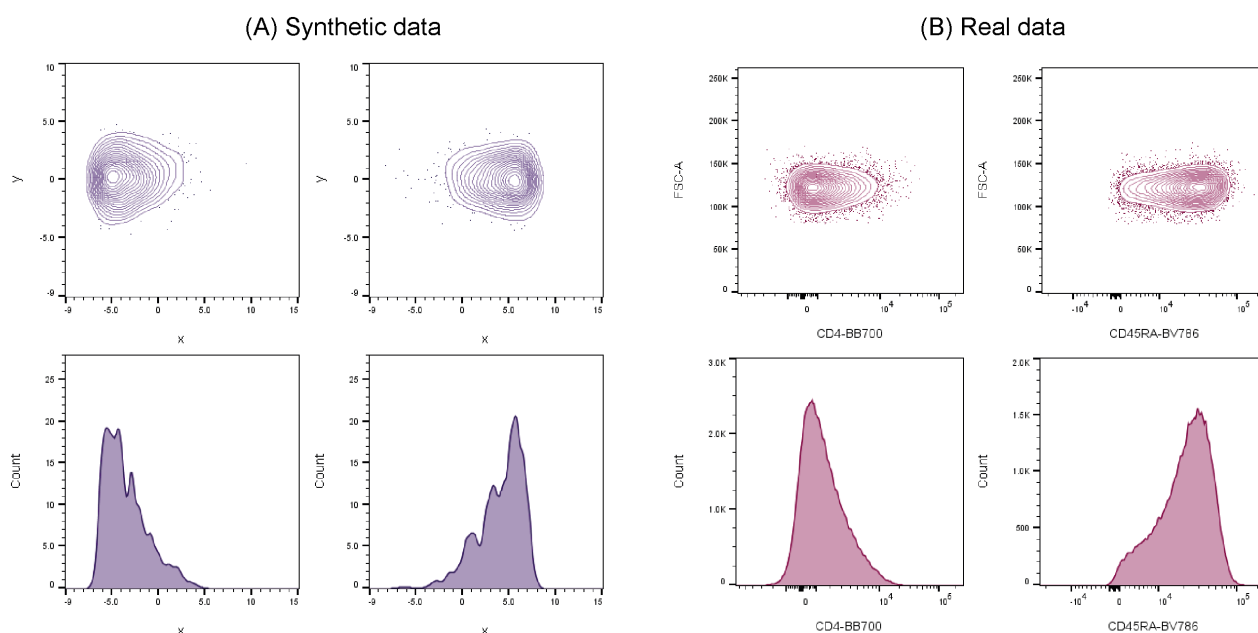


Figure 1. Inter-comparison between synthetic and real flow cytometry data showing clear similarities among cell distribution properties.

Synthetic datasets are used in this research because: (1) they simplify the complexities of real-world data, enabling the separation of interacting factors that cloud the understanding and assessment of automated software, (2) they provide a ground truth that allows measurement accuracy to be explicitly determined (something that is difficult to achieve using real cell data), and (3) they overcome challenges in the acquisition of biological samples related to time, cost, scarcity of rare samples, and data privacy concerns.

Within the research reported here, the novel synthetic datasets containing controlled separation between clusters with normal or non-normal probability distributions are applied to a selection of flow cytometry computational tools that utilise different classes of clustering algorithms. We compare the performances of these different software using accuracy and repeatability evaluation metrics for showing trends in performance between software when analysing clusters with specific degrees of separation, and with skewed cell populations.

This research is intended to provide flow cytometry users in the biomanufacturing community with a better understanding of the characteristics, opportunities and limitations of automated data analysis software, ideally leading to enhanced confidence in the data quality of cell characterisations. In addition, it shows how a framework for benchmarking toolsets can be specifically designed for selection/validation of automated data analysis software.

2. Materials and Methods

2.1. Datasets

In order to perform a fair comparison between different automated data analysis software, synthetic flow cytometry reference datasets were designed and generated (as described in [21]). Out of the commonly recognised data characteristics or potential statistical attributes identified, we targeted the separation and the skew characteristics to control and modify in our datasets, because these properties had not been addressed in previous work and/or the designs had not been approached in a systematic manner. To retain the focus on these properties, non-target characteristics such as cluster sizes and the number of dimensions were kept constant, and the element of noise relating to real data was excluded.

2.1.1. Separation Dataset

The purpose of these datasets was to evaluate software performance in identifying and partitioning cell populations as the clusters came close together.

Separation datasets were prepared using the R *clusterGeneration* package [22] with the following parameters:

- Number of clusters: 2 or 3,
- Cluster size: 1000 points per cluster,
- Separation index (SI) values: from -0.3 to $+0.3$, at 0.1 intervals, and
- Cluster covariance matrices: eigenvalues between 1 and 5.

This approach generated datasets with different degrees of separation between neighbouring clusters ranging from well separated to merged. Nine random normally distributed cluster replicates were generated at each SI value. Parameters were selected to give a range of variability in the diameter and shape of clusters similar to those seen in exemplar real flow cytometry data. Datasets were converted to FCS 3.1 format using the R package *flowCore* [23].

2.1.2. Skew Dataset

The purpose of this skew dataset was to evaluate software performance in identifying and partitioning cell populations as the clusters displayed different levels of non-normal distributions.

Skew datasets were built in multiple stages. First, individual skew clusters were prepared with the function *rmsn* in the package *sn* [24], using the following parameters:

- Number of clusters: 1 (clusters later joined together),
- Cluster size: 1000 points per cluster,
- Mean vector: $[0, 0]$,
- Covariance matrix: values between 1 and 5, and
- α skew value: values between 2.5 and 10, at intervals of 2.5.

of which the α parameter regulated asymmetry. Likewise random cluster replicates were generated at each skew direction (left and right) along the x -axis.

During cluster generation, it was found that applying the skewing parameter (α) caused the diameter of the elliptical cluster to reduce along the x -axis. To compensate for this, clusters were elongated to obtain a pre-skew diameter using the R package *scales* [25]. The skewness of the clusters before and after rescaling were identical (measured using the R package *psych* [26]) determined by the asymmetry around the mean remaining unchanged.

Next, two clusters were joined together. A new level of complexity was introduced compared to normally distributed clusters because asymmetric clusters could be orientated in three ways: head to head, head to tail, and tail to tail (assuming the skew is introduced only along the x -axis). Clusters with the same α skew values were paired together (i.e., clusters with different skews were not combined).

Finally, one cluster was shifted further away from the other through vector operations in R. The distance between two clusters was measured with the *clusterGeneration* package [22]; datasets with a SI value between -0.25 and -0.15 were selected for further processing. Files were again converted to FCS 3.1 standard using *flowCore* [23].

This approach generated a library of two-cluster synthetic datasets in two dimensions with 1000 datapoints per cluster, with different levels of skew and skew-direction pairs.

2.2. Software Runs

The synthetic datasets were processed through six flow cytometry automated data analysis software (Table 1), each of which implement different unsupervised clustering algorithms: Flock2 [27] (via ImmPort Galaxy [28]), flowMeans [29], FlowSOM [30], PhenoGraph [31] (R implementation [32]), SPADE3 [33,34] and SWIFT [35,36].

It is recognised that many supervised techniques for automated cell population identification are available (e.g., FlowDensity [37]); however, these tools have not been included

in this study because a significantly different approach in actual methodology of synthetic dataset design/application would be needed, mainly the need for extensive pre-training and training datasets containing meta-labelled data, and specific design of testing datasets.

Table 1. Description of computational tools used in this study.

Computational Tool	Description	Reference
Flock2	FLOW Clustering without K; grid-based density clustering algorithm, where the data are divided into hyper-regions, then dense regions are identified, merged and points assigned to their nearest centroids.	[27]
flowMeans	<i>k</i> -Means-based clustering that allows multiple clusters to model a single population, with overlapping clusters later being merged.	[29]
FlowSOM	A workflow that reads the data, builds a self-organising map (SOM), builds a minimal spanning tree then computes a meta-clustering output.	[30]
PhenoGraph	Constructs a <i>k</i> nearest neighbour graph from high-dimensional data, then uses the Louvain community detection algorithm to partition the graph into sub-populations. Spanning-tree progression analysis for density-normalised events; performs deterministic density-dependent down-sampling, then <i>k</i> -means -based clustering, followed by minimal spanning tree construction. A tree partitioning algorithm aids semiautomated interpretation of data.	[31]
SPADE3	Scalable Weighted Iterative Flow-clustering Technique; Gaussian mixture model-based clustering, followed by splitting and merging steps to obtain final clusters that are unimodal but not necessarily Gaussian.	[33,34]
SWIFT		[35,36]

2.3. Statistics and Performance Evaluation Metrics

Methods used for statistical analysis included the mean, sample standard deviation, and coefficient of variation (CV).

The software outputs were recorded, and the absolute difference between cell populations of cluster 1 to the reference value was calculated in percentage terms, as in Equation (1).

$$\text{Difference to reference \%} = \frac{|A - B|}{\text{Total events}} \times 100 \quad (1)$$

where *A* is the reference cluster 1 count, and *B* is the software cluster 1 count.

3. Results

3.1. Output Number of Clusters

We assessed the performance of the six automated data analysis software, each of which implement different clustering algorithms to identify and quantify cells: Flock2, flowMeans, FlowSOM, PhenoGraph, SPADE3, and SWIFT (density-based, *k*-means, self-organising map, *k*-nearest neighbour, deterministic *k*-means, and model-based clustering, respectively).

We first investigated whether the software could partition the datasets to give the same number of clusters originally designed into them. We found that returning the desired number of clusters was straightforward for tools such as flowMeans, where the input number of clusters (*k*) directly determined the output. Obtaining the desired number of clusters from other software was more complex. In SWIFT, the input *k* served as an initial estimate which sometimes varied from the final output cluster number after subsequent cluster splitting and merging processing steps. In SPADE3, the default user settings automatically over-clustered the data into a minimum spanning tree with hundreds of nodes, with a subsequent ‘semi-automated’ feature to suggest tree partitioning to the user.

Here, the tree partitioning step was applied until the desired number of clusters were produced. PhenoGraph, and occasionally Flock2 and SWIFT, tended to over-cluster the data, so additional manual steps were performed to merge sub-clusters together.

In general, the manual workload increased in proportion to the number of clusters generated by a software above the desired amount, illustrating a paradox of increased human intervention in a supposedly automated process designed to reduce operator variation. We also found that flowMeans and FlowSOM did not permit outputs of two clusters, so processing of two-cluster datasets returned a minimum cluster number of three, thus again requiring a manual merging step.

Overall, strategies to obtain the desired output number of clusters varied significantly between different software, with some requiring repeated tuning of input parameters and/or post-clustering manual interpretation steps, suggesting a high level of operator training required, as opposed to casual use.

3.2. Clustering Characteristics

The different software tools tested here all utilised different clustering algorithms, and certain data partitioning characteristics became particularly noticeable with overlapping clusters as the data became unstructured. Reference cluster designs are depicted in Figures 2A and 3A, along with the raw software clustering outputs, before manual intervention was performed to merge sub-clusters together from, e.g., Flock2, flowMeans, FlowSOM and PhenoGraph. Scatterplots of the software clustering results show how neighbouring clusters from Flock2 and flowMeans were separated with hard straight line boundaries often radiating from a central region (Figures 2B,C and 3B,C), whereas divisions among FlowSOM, PhenoGraph and SPADE3 clusters resembled meandering twisting lines that had echoes of underlying merged sub-clusters (Figures 2D–F and 3D–F). Clusters from SWIFT had softer boundaries, with the fitted Gaussian models visible that slightly overlap each other (Figures 2G and 3G).

3.3. Two-Cluster Separation

To assess the performance of software as cell populations come closer together, synthetic two-cluster datasets were generated with multiple replicates at each separation index condition (as described in Section 2.1.1).

While clusters remained separate and distinct with a $SI \geq 0.1$, all software outputs were similar to the reference value (differences ranged from 0.01% to 0.97%), and strong repeatability was observed (all standard deviations below 0.8). However, as the two clusters came closer together and the SI approached and fell below 0.0, all six software platforms displayed a decrease in performance; the differences between the software values and the reference value widened, and repeatability deteriorated as demonstrated by the extent of the error bars (Figure 4). The critical SI region appeared to be around -0.1 , and any further overlapping of clusters resulted in sharp reductions in software performance and erratic outputs. To place this in the context of real data, the identification of chimeric antigen receptor (CAR)-T cells (e.g., on the basis of the CD19 protein) routinely requires the analysis of less well-separated clusters that fall into this SI region of -0.1 [38]. Overlapping clusters appeared to have the most detrimental effect on Flock2 performance, with differences to the reference value widening from $(3.0 \pm 4.1)\%$ at $SI -0.1$ to $(11.9 \pm 9.6)\%$ at $SI -0.2$. flowMeans showed similar trends of reduced performance, with difference to reference of $(6.1 \pm 4.0)\%$ at $SI -0.1$ and $(9.6 \pm 4.3)\%$ at $SI -0.2$. In contrast, the smaller differences in SWIFT outputs to reference from $(1.4 \pm 0.73)\%$ to $(5.7 \pm 2.6)\%$ at $SI -0.1$ and -0.2 , respectively, indicated somewhat better detection of overlapping normally distributed cell populations. However, SWIFT was not able to return two clusters at $SI -0.3$.

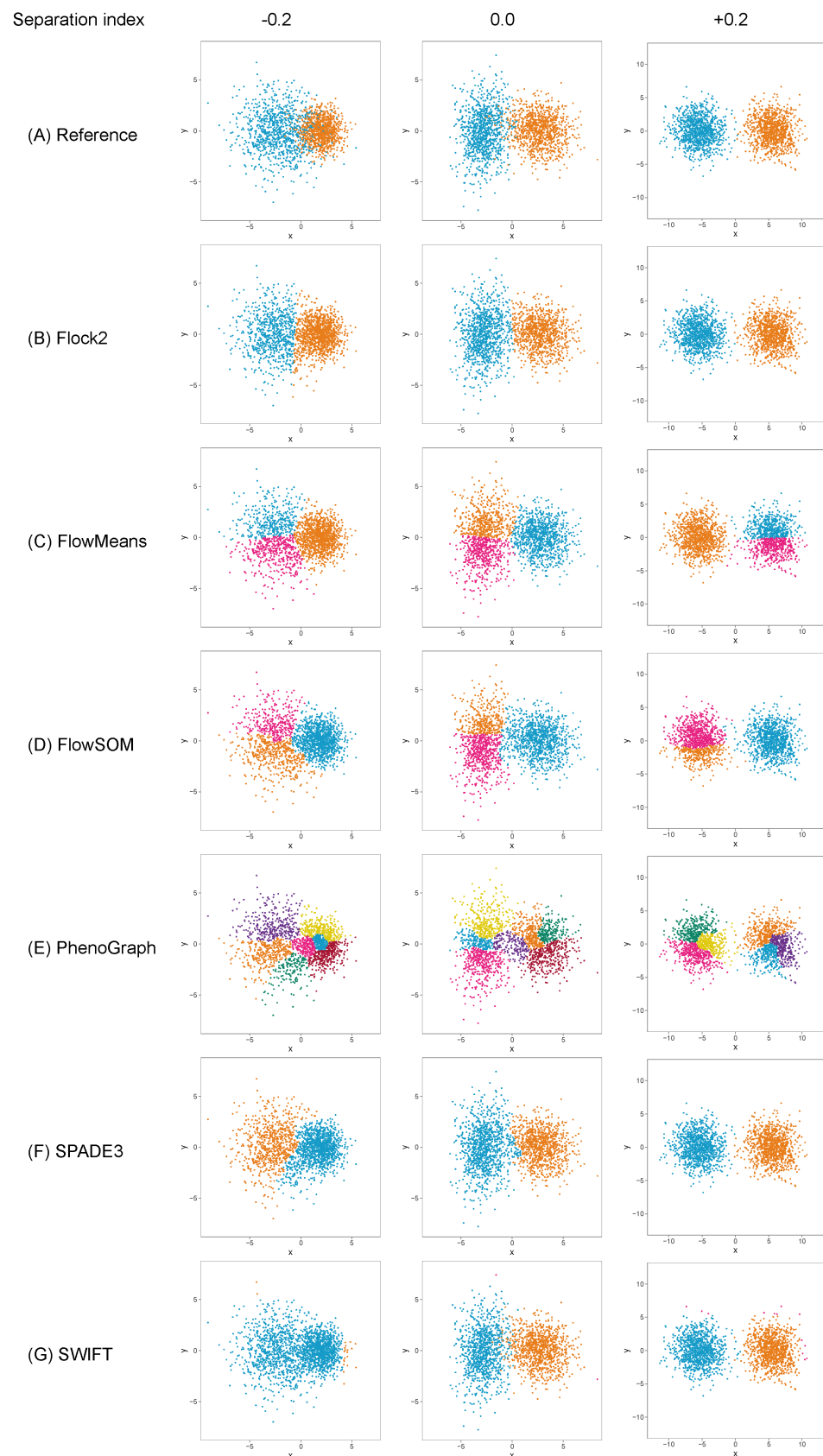


Figure 2. Clustering examples from different software on a two-cluster synthetic flow cytometry dataset with different degrees of separation between clusters.

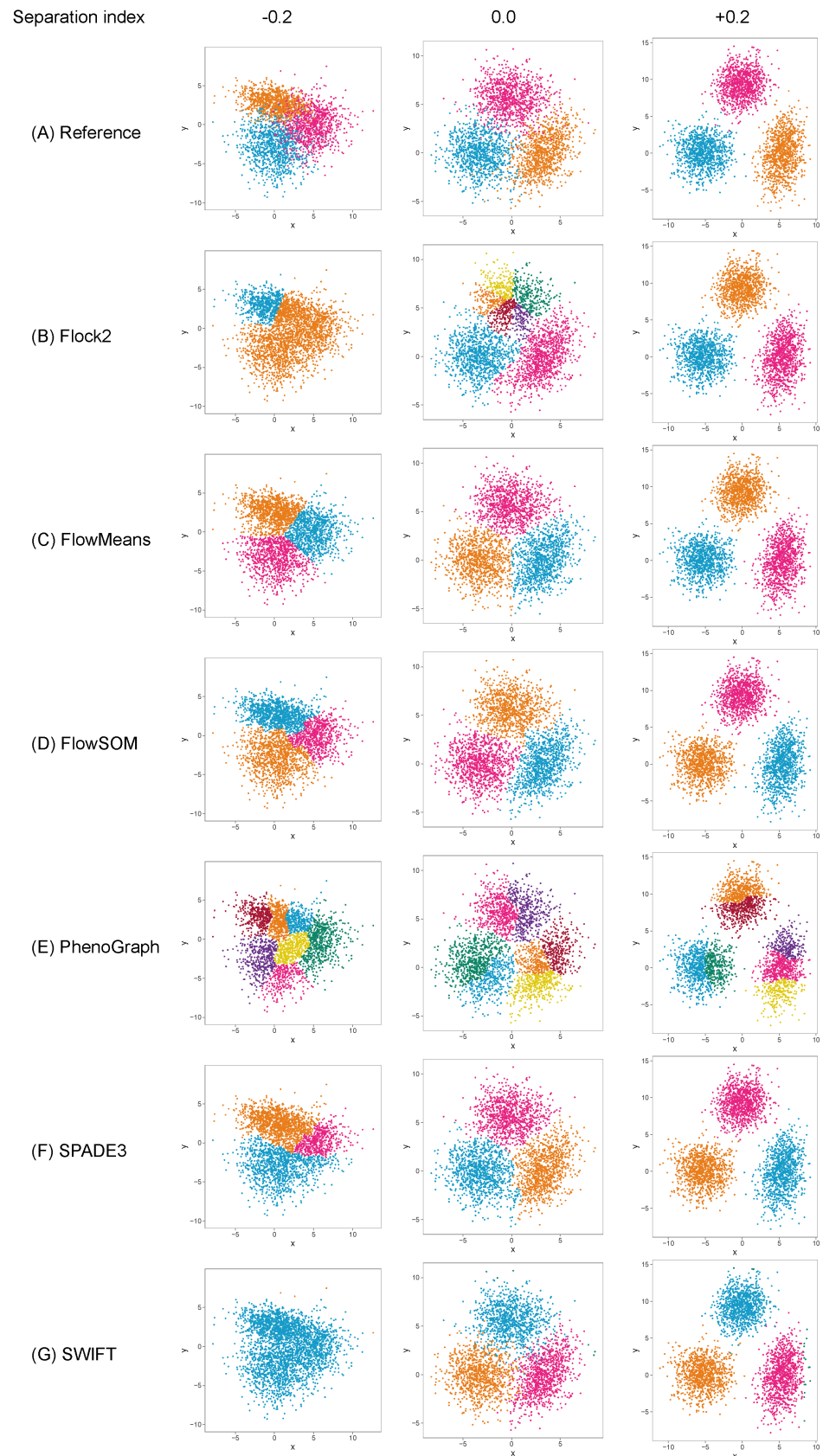


Figure 3. Clustering examples from different software on a three-cluster synthetic flow cytometry dataset with different degrees of separation between clusters.

Overall, application of the synthetic two-cluster separation dataset revealed that SWIFT performed better compared to FlowSOM, followed by SPADE3 and PhenoGraph in terms of accuracy and repeatability.

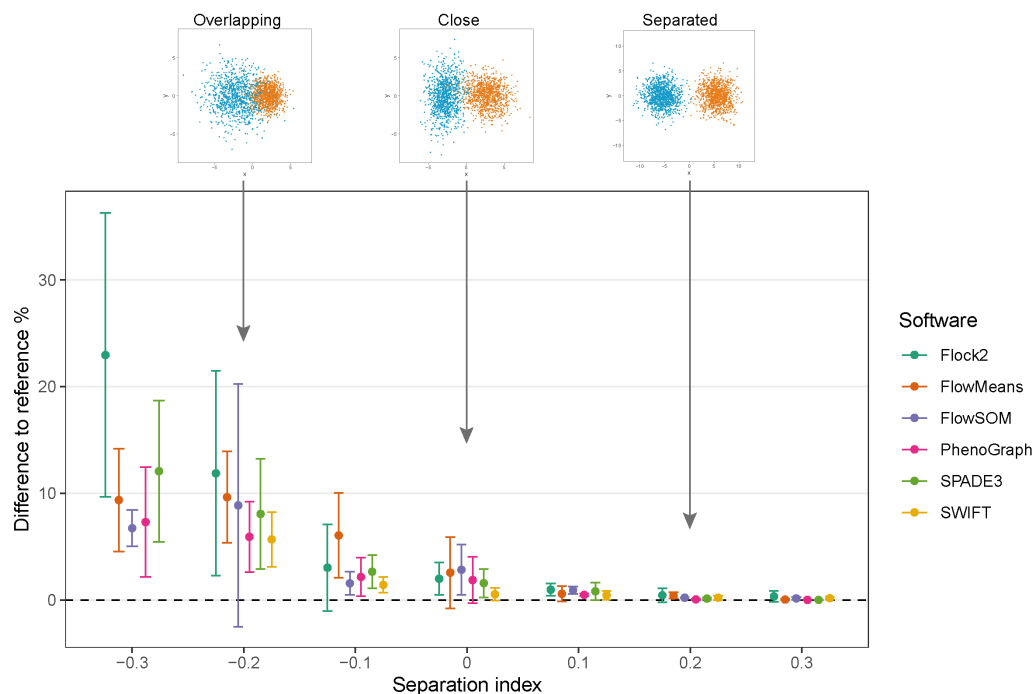


Figure 4. Performance of different software with a two-cluster separation dataset.

3.4. Three-Cluster Separation

Evaluation of the effect of cluster separation on software performance was extended by introducing another cluster to the dataset. The three-cluster dataset added an additional level of complexity as the software now had to make two partitions in the dataset rather than one. Having three clusters also negated issues such as FlowSOM giving a minimum cluster of three for the two-cluster dataset. After causing each software to return three clusters, the number of points per cluster was recorded and the population of cluster 1 was arbitrarily selected to compare against the reference percentage of population value of 50%.

The results displayed similar trends in accuracy and precision to the two-cluster dataset (Figure 5). All of the software maintained good accuracy and repeatability at $SI \geq 0$, with the exception of FlowSOM at SI 0.1, which displayed lower performance than others. As the SI decreased below 0, software performance again began to deteriorate. The reduction in performance for all software was again particularly noticeable from SI -0.1 to -0.2 . Below SI -0.2 , the deterioration of performance appeared to plateau for flowMeans, PhenoGraph and SPADE3. Given that it showed consistently smaller differences to the reference value at $SI \geq 0$ than other software, flowMeans appeared to be less affected by overlapping clusters; however, whether this was a merit of the software or a consequence of ‘random’ equal partitioning of the dataset will require further investigation. Flock2 did not identify three clusters at SI -0.2 , and SWIFT at SI -0.3 , further highlighting regions of the separation index dataset where clusters became difficult to resolve. Again, it is noted that three-cluster partitioning is prevalent in manual cell analysis.

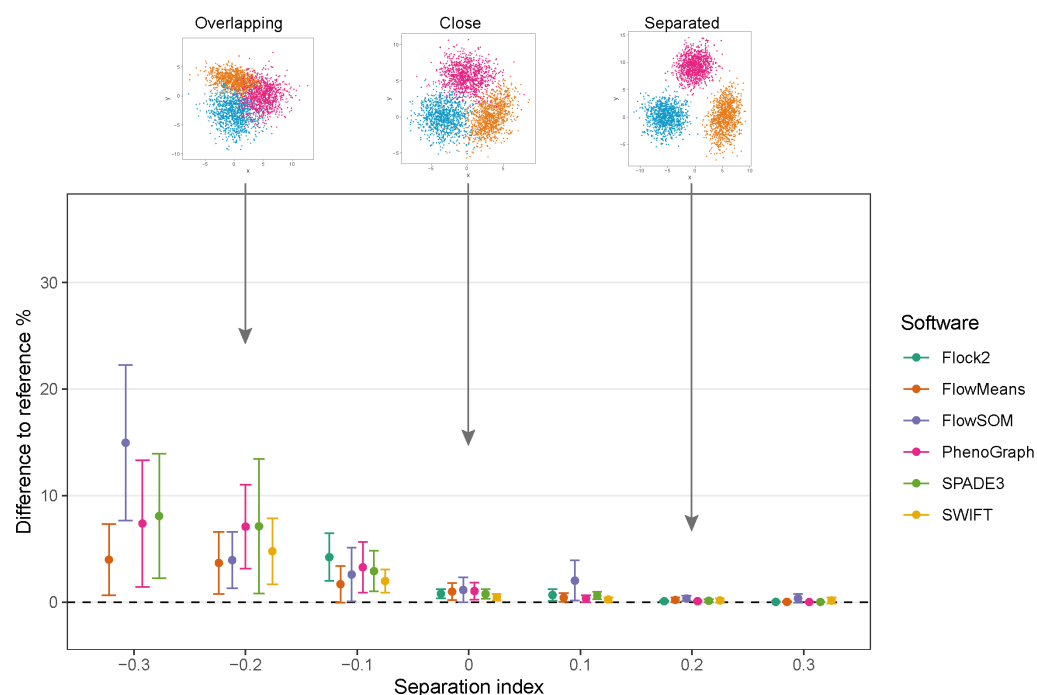


Figure 5. Performance of different software with a three-cluster separation dataset.

3.5. Skew

To understand whether the behaviours of software were limited to clusters with normal distributions, datasets containing clusters ranging from normal symmetrical to more asymmetrical skewed distributions were generated and processed through the software. Initial runs were performed on skew cluster pairs with a tail-to-tail orientation used here as an exemplar of real flow cytometry data.

The results showed that, once again, different software returned different clustering outputs and partitioning characteristics from the same dataset (Figure 6). Obtaining the desired output number of clusters, two, was straightforward with Flock2, flowMeans, SPADE3 and SWIFT. FlowSOM gave a minimum output of three clusters, resulting in the appearance of a horizontal bisect of one of the two populations. PhenoGraph outputs partitioned the data into approximately eight clusters as a result of the k value that was selected as a compromise between excessive manual intervention and long computational run times (see Supplementary Materials). It is noted that the PhenoGraph algorithm may not be appropriate for analysis of the low-dimensional datasets applied here, however previous work have tested PhenoGraph performance using artificial two-dimensional data [39], and its inclusion in this comparison study here remains useful for users.

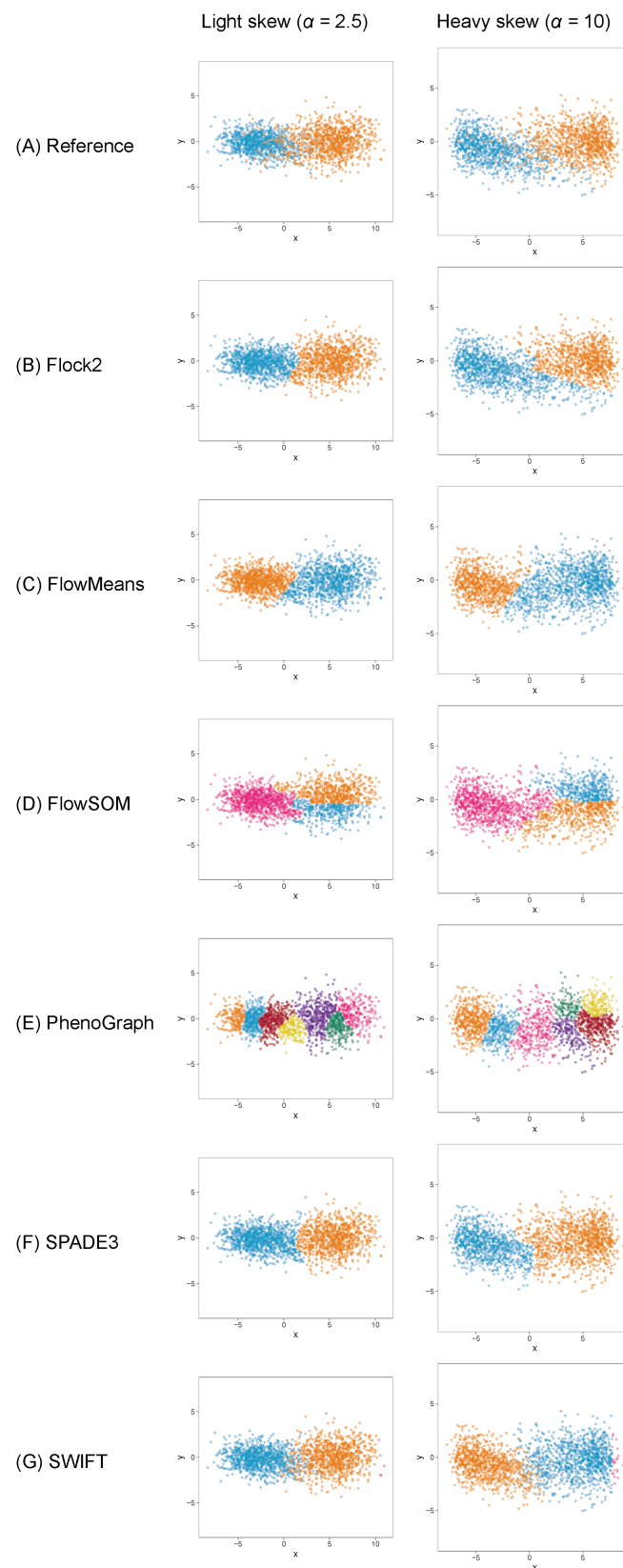


Figure 6. Clustering examples from different software on a two-cluster dataset with skew distributions. Two levels of skew are shown, light skew ($\alpha = 2.5$) and heavy skew ($\alpha = 10$), with cluster orientations all facing tail to tail.

Software responded to increasing levels of skew in different ways. In clusters with heavy skew, Flock2, FlowSOM and SPADE3 appeared to partition the data in a more similar manner to the reference dataset compared with flowMeans, PhenoGraph and SWIFT (Figure 6). In this tail-to-tail configuration, Flock2 outputs showed improved accuracy and repeatability as the levels of skew increased, going from a difference to reference of $(23.5 \pm 16.1)\%$ at no skew ($\alpha = 0$) to $(4.2 \pm 3.0)\%$ at heavy skew ($\alpha = 10$) (Figure 7). The opposite effect was observed for PhenoGraph, with the gap to reference widening from $(5.2 \pm 3.9)\%$ at skew $\alpha = 0$ to $(9.7 \pm 8.4)\%$ at skew $\alpha = 10$. In comparison, other software outputs showed no significant differences in performance as illustrated in Figure 6. A weak trend was observed for SPADE3 to have better accuracy and repeatability as the level of skew in the datasets increased, and the opposite trend (slight decrease in performance) was observed for SWIFT (Figure 7).

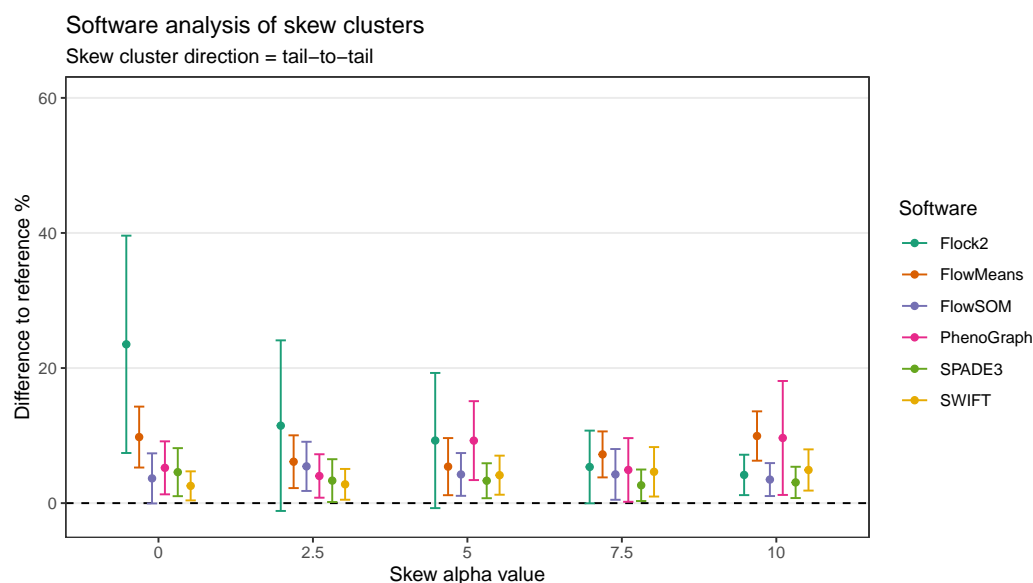


Figure 7. Performance of different software on a dataset with skew cluster orientations facing tail to tail.

3.6. Skew Orientation

It was thought that as well as the level of skewness, the orientation of skew clusters to each other could be a factor affecting a software's ability to identify cell populations. To investigate this further, the two-cluster skew dataset (initially orientated tail to tail), was extended to include cluster pairs facing both head-to-head and head-to-tail directions (Figure 8). Again, it was seen that whilst most software were able to return two clusters, FlowSOM returned three clusters, and PhenoGraph overclustered the data.

The extension of the skew dataset revealed SWIFT to be the software most affected by skew clusters. In the head-to-head configuration, the gap to reference declined from $(2.6 \pm 2.2)\%$ at skew $\alpha = 0$ to $(35.7 \pm 21.6)\%$ at skew $\alpha = 7.5$ (Figure 9). Furthermore, SWIFT failed to return any output at skew $\alpha = 10$. The head-to-head pairings also showed flowMeans decreased in performance with increasing skew, with difference to reference going from $(9.8 \pm 4.5)\%$ at skew $\alpha = 0$ to $(18.0 \pm 4.5)\%$ at skew $\alpha = 10$.

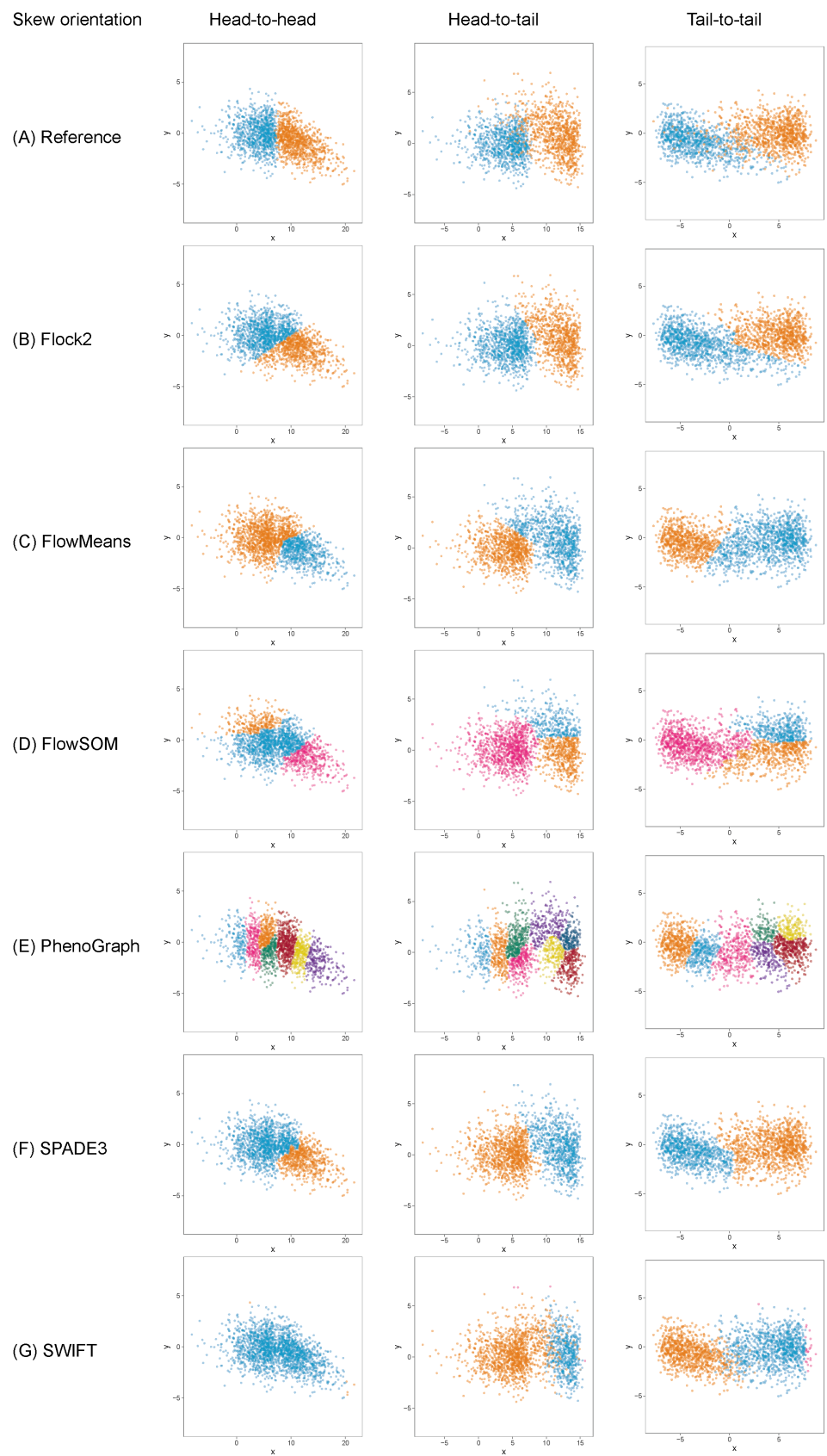


Figure 8. Clustering examples from different software on a two-cluster dataset with skew pairs facing different orientations. All clusters shown with heavy skew ($\alpha = 10$).

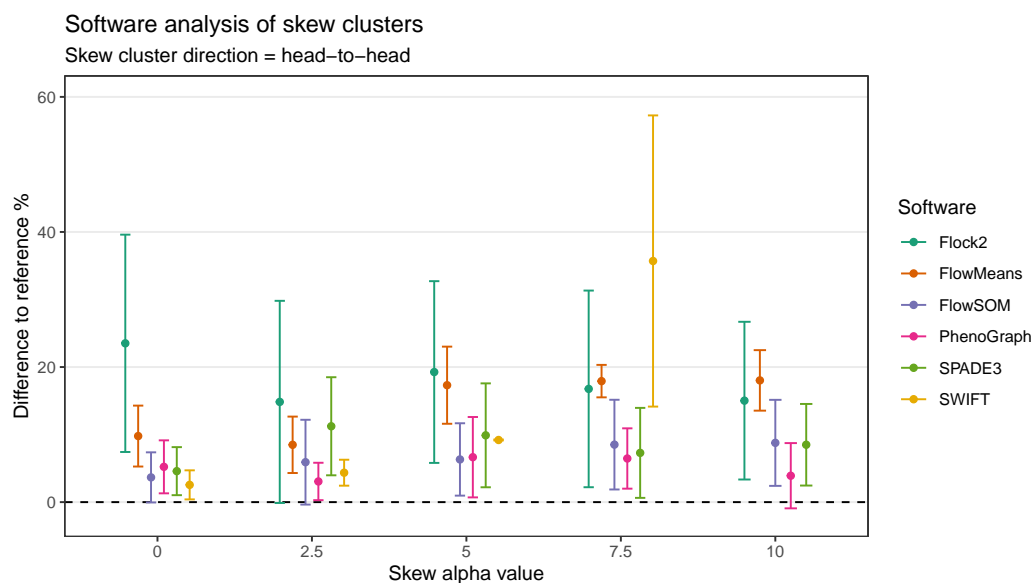


Figure 9. Performance of different software on a dataset with skew cluster orientations facing head to head.

Comparison across all software suggested that FlowSOM and SPADE3 were least affected by skew distributions, both outperformed Flock2 and flowMeans in terms of accuracy and repeatability.

In the head-to-tail orientation, SWIFT's performance was noticeably lower than other software at every level of skew above 0 (Figure 10). For instance, the difference to reference of $(21.3 \pm 3.0)\%$ at skew $\alpha = 7.5$ was worse than the average of all other software ($7.5 \pm 3.8\%$). This suggested that the strategy SWIFT utilises to fit data to Gaussian distributions followed by splitting and merging steps may be challenged by the processing of non-Gaussian distributions.

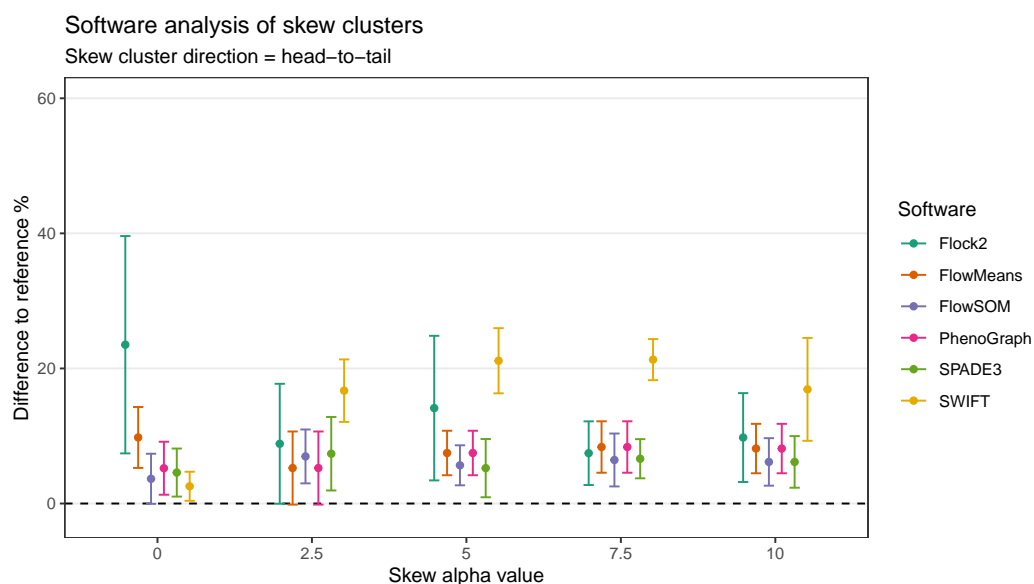


Figure 10. Performance of different software on a dataset with skew cluster orientations facing head to tail.

An alternative visualisation of the results from the skew dataset runs suggests that most of the software tested showed a decline in accuracy and repeatability as the orientation shifted from tail to tail, to head to tail and then head to head, respectively (Figure 11). This

pattern was generally observed at all levels of skew tested. The changes in performance was likely due to the reduction in the density of events in between the two clusters moving between one orientation to the other, i.e., the higher density of interface events in the head-to-head orientation made data partitioning more difficult. An interesting exception to this pattern was observed with PhenoGraph, where analysis of tail-to-tail skew clusters appeared to slightly reduce in accuracy and repeatability compared with the head-to-head orientated skew clusters. This was possibly because of characteristics of the PhenoGraph algorithm, or more likely that the significant manual intervention required to merge output clusters together to achieve final outcomes artificially improved PhenoGraph results.

Software analysis of skew clusters

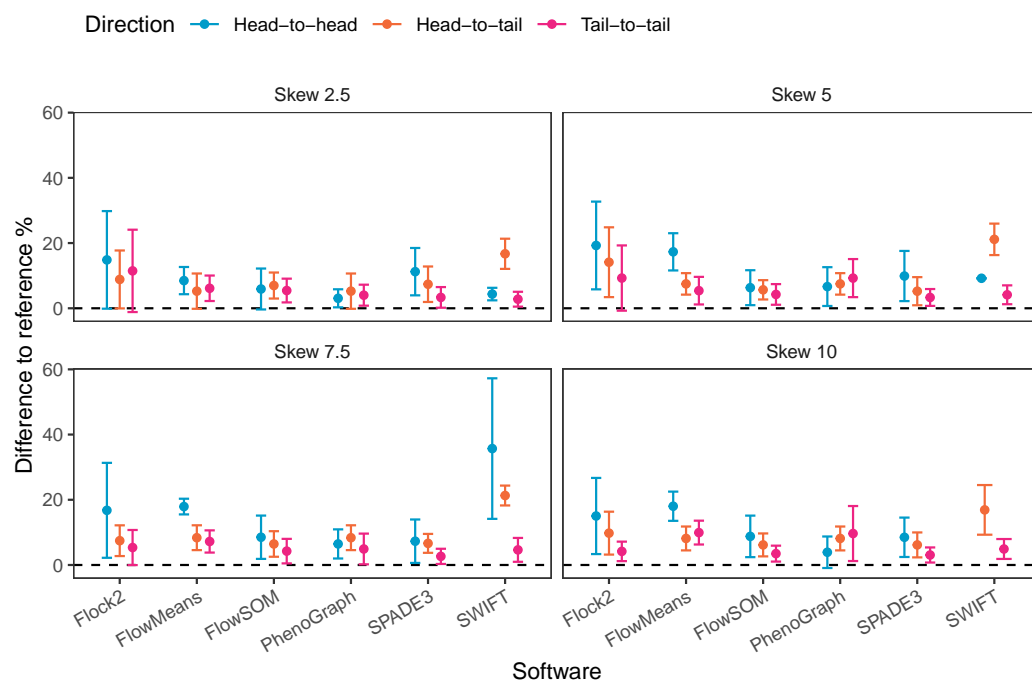


Figure 11. Performance of different software on a dataset with skew cluster orientations facing head to head, head to tail, and tail to tail.

Taken together, automated analysis of our synthetic skewed dataset revealed the effects of skew on software performance were largely software dependent, and affected different classes of clustering algorithms in varying ways. Software that model Gaussian distributions onto data were the least well performing (flowMeans and SWIFT). Density-based clustering software appeared to be unaffected by skew characteristics in the data (Flock2). FlowSOM, SPADE3 and PhenoGraph performed well against other software tested here, potentially because they implement overclustering steps that break up the data into smaller populations that each differ in skew properties from the main major population.

4. Discussion

Characterisation of ATMPs by automated flow cytometry data analysis software have the potential to improve the quality, repeatability, and robustness of biomanufacturing processes by reducing operator variation as a function of subjective manual gating of clustered data. However, the lack of clarity in how these software derived outputs from inputs, coupled with the absence of toolsets for software validation and standardisation, potentially restricts their implementation by the manufacturing community. In addition, it presents challenges from a clinical and regulatory perspective.

Our previous work on the inter-comparison between synthetic and a real dataset showed clear correlation among cell distribution characteristics examples [21]. Conse-

quently, for this particular cross-platform comparison we are confident that the synthetic data mirrors, to an appropriate level, the key characteristics of low dimensionality cluster data, demonstrates design flexibility and application, and allows for traceable benchmarking (absolute accuracy and repeatability), without the further need to run the platforms through further real data.

In this study, synthetic datasets have been designed and applied to test the performance of six automated flow cytometry cell population identification computational tools. Our use of synthetic datasets with controlled distances between clusters demonstrated similar patterns of behaviours between different software, in which accuracy and repeatability deteriorated as clusters came closer together, particularly below the separation index value of -0.1 . These software responses were expected given that overlapping clusters change from multi-modal to unimodal distributions, progressively becoming one large cluster with merged cell populations. The skew datasets implemented here identified considerable variation in outputs between software when processing non-Gaussian distributed clusters, reflecting the different mathematical approaches employed by software to identify cell populations.

Among the six automated tools assessed here, the SWIFT algorithm was found to display better accuracy and repeatability compared to other tools as normally distributed clusters began to overlap and their separation index shifted below 0. However, when assessed further with skewed clusters, SWIFT performance noticeably declined more than others as the skew levels increased. Insights such as these can give operators unfamiliar with computational tools and algorithms a deeper understanding of the potential optimal working ranges of these tools, and the variations in performance that can arise between them depending on the data structures. Furthermore, it could support upstream assay design to ensure data outputs are fit for automated analysis, such as switching to fluorophores leading to more optimised separation, or acquisition settings.

The synthetic dataset approach applied in this study to evaluate automated cell population identification tools extends on, but cannot be directly compared with findings from previous comparison studies, because of the differences in datasets (synthetic and real world) and dataset characteristics used. For example, studies have previously identified FlowSOM as high performing based on high-dimensional datasets [12]; however, in this study, SWIFT outperformed FlowSOM in the low-dimensional, normally distributed dataset, although further testing in the presence of skewed clusters saw SWIFT performance deteriorate.

Compared with previous software comparison studies, the datasets applied here reduce the dependence on narrow cell model examples. Further strengths of this approach include the use of measurable distances between clusters through the separation index, as well as controllable skew parameters, with the benefit of allowing computational tools to be tested one factor at a time, on controlled sets of criteria not feasibly generated from experimental conditions. Of note, the synthetic datasets allow comparisons of software outputs away from subjective manually gated reference values that lack a 'ground truth' thus providing explicit statements of accuracy and repeatability.

This study specifically targeted the variation arising from data analysis within the flow cytometry analytical process. Upstream sources of variation from starting materials, sample processing, and instrumentation would require separate comparison studies designed around those factors as variables (e.g., conditions such as lysis, wash, and staining) and with the data analysis software tool kept constant. With regards to the relevance of this data to biological samples analysis, the synthetic datasets here have been designed with essential properties that simulate their equivalent biological counterparts. Therefore, software runs that fail on encountering such data characteristics would directly infer on the (lack of) credibility of results from similar biological samples.

A recognised limitation of this work is that the number of markers simulated is lower than those in real data (usually > 3 -colour panels) because a priority in this study has been to understand and benchmark how algorithms behave with two or three clusters before

introducing further complexities into the datasets. Noting the successful referencing and correlation study we have already completed between synthetic and real data [21], overall, real data have been excluded from this initial research because they are significantly more complex, containing sources of variation from upstream processes and noise components that cannot be controlled to transparently understand the ‘black box’ nature of the algorithms investigated. Additionally, it is very difficult to achieve absolute cell counts for real data, so defining measurement accuracy (a critical component of this study) would not be possible. This research here has applied clearly defined synthetic datasets to establish the base functionality of software at lower numbers of parameters before escalating to higher-dimensional datasets (i.e., we cannot run before we can walk). Having achieved this, building more complex datasets is the next key area for further work, and once at that stage, further comparisons between real datasets will illustrate even greater relevance.

The results presented here open up further work to explore more data properties in synthetic dataset design, such as inclusion of more cell populations, higher dimensionality, noise parameters, and in particular, rare cell populations—assessment of which will be a subject for further work within our research. To address the potential for a more heterogeneous cell population mix rather than the homogeneous ones depicted here, further work could model cell subsets within the bulk-component of the skewed population with changing phenotypes (e.g., stem cells undergoing differentiation, T cells in response to cytokine activation), in line with the escalation of various components of complexity within the synthetic dataset design.

Overall, the results of this study suggest that benchmarking of automated flow cytometry software platforms will be possible with a high level of testing integrity using synthetic cluster datasets. The goal of this work was initially to enable biomanufacturers to make better informed decisions about whether or not to implement automated data analysis tools in their workflow instead of/in addition to manual gating methods, based on their own cytometry data—although it is clear that it is also relevant to the clinical community and would potentially impact regulatory science.

Where advanced analysis methods are deemed necessary, the clustering characteristics of different analysis tools illustrated here will facilitate the selection of ones that are fit for purpose. For users, these toolsets can be used to validate and verify installed software and confirm that working ranges match the specifications of their own data. For regulators inspecting process validation documentation, the inclusion of these datasets to provide assurances in automated cell characterisation measurement processes would potentially be desirable. There is potential for the development of synthetic digital reference materials to provide assurances in advanced analytical methods, leading to enhanced measurement confidence in ATMPs.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/ijms23063224/s1>.

Author Contributions: Conceptualisation, M.C., J.J.C., J.B. and J.P.; methodology, M.C.; software, M.C.; validation, M.C.; formal analysis, M.C.; investigation, M.C.; resources, J.P.; data curation, M.C.; writing—original draft preparation, M.C.; writing—review and editing, M.C., J.J.C., R.J.T., J.B. and J.P.; visualisation, M.C.; supervision, J.P.; project administration, J.P.; funding acquisition, J.B. and J.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was kindly supported by the EPSRC/MRC Doctoral Training Centre for Regenerative Medicine at Loughborough University (EP/L105072/1), and LGC Ltd.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. EMA. Guideline on Quality, Non-Clinical and Clinical Requirements for Investigational Advanced Therapy Medicinal Products in Clinical Trials. 2019. Available online: <https://www.ema.europa.eu/en/guideline-quality-non-clinical-clinical-requirements-investigational-advanced-therapy-medicinal> (accessed on 2 July 2021).
2. FDA. Chemistry, Manufacturing, and Control (CMC) Information for Human Gene Therapy Investigational New Drug Applications (INDs). 2018. Available online: <https://www.fda.gov/regulatory-information/search-fda-guidance-documents/chemistry-manufacturing-and-control-cmc-information-human-gene-therapy-investigational-new-drug> (accessed on 2 July 2021).
3. Campbell, J.D.; Fraser, A.R. Flow cytometric assays for identity, safety and potency of cellular therapies. *Cytom. Part B Clin. Cytom.* **2018**, *94*, 569–579. [[CrossRef](#)]
4. Arcidiacono, J.A.; Bauer, S.R.; Kaplan, D.S.; Allocca, C.M.; Sarkar, S.; Lin-Gibson, S. FDA and NIST collaboration on standards development activities supporting innovation and translation of regenerative medicine products. *Cytotherapy* **2018**, *20*, 779–784. [[CrossRef](#)]
5. British Pharmacopoeia. Guidance on the Application of Flow Cytometry for the Cell and Gene Therapy Community. Draft Document. 2021. Available online: <https://www.pharmacopoeia.com/bp-consultations/flow-cytometry> (accessed on 10 June 2021).
6. Maecker, H.T.; Rinfret, A.; D'Souza, P.; Darden, J.; Roig, E.; Landry, C.; Hayes, P.; Birungi, J.; Anzala, O.; Garcia, M.; et al. Standardization of cytokine flow cytometry assays. *BMC Immunol.* **2005**, *6*, 13. [[CrossRef](#)]
7. Grant, R.; Coopman, K.; Medcalf, N.; Silva-Gomes, S.; Campbell, J.J.; Kara, B.; Braybrook, J.; Petzing, J. Understanding the contribution of operator measurement variability within flow cytometry data analysis for quality control of cell and gene therapy manufacturing. *Measurement* **2020**, *150*, 106998. [[CrossRef](#)]
8. Grant, R.; Coopman, K.; Medcalf, N.; Silva-Gomes, S.; Campbell, J.J.; Kara, B.; Braybrook, J.; Petzing, J.N. Quantifying operator subjectivity within flow cytometry data analysis as a source of measurement uncertainty and the impact of experience on results. *PDA J. Pharm. Sci. Technol.* **2021**, *75*, 33–47. [[CrossRef](#)]
9. Cheung, M.; Campbell, J.J.; Whitby, L.; Thomas, R.J.; Braybrook, J.; Petzing, J.N. Current trends in flow cytometry automated data analysis software. *Cytom. Part A* **2021**, *99*, 1007–1021. [[CrossRef](#)]
10. Saeys, Y.; Van Gassen, S.; Lambrecht, B.N. Computational flow cytometry: Helping to make sense of high-dimensional immunology data. *Nat. Rev. Immunol.* **2016**, *16*, 449–462. [[CrossRef](#)]
11. Aghaeepour, N.; Finak, G.; Hoos, H.; Mosmann, T.R.; Brinkman, R.; Gottardo, R.; Scheuermann, R.H.; Gottardo, R.; Scheuermann, R.H. Critical assessment of automated flow cytometry data analysis techniques. *Nat. Methods* **2013**, *10*, 228–238. [[CrossRef](#)]
12. Weber, L.M.; Robinson, M.D. Comparison of clustering methods for high-dimensional single-cell flow and mass cytometry data. *Cytom. Part A* **2016**, *89*, 1084–1096. [[CrossRef](#)]
13. NIST. NIST Flow Cytometry Standards Consortium. 2020. Available online: <https://www.nist.gov/programs-projects/nist-flow-cytometry-standards-consortium> (accessed on 2 July 2021).
14. FDA. Software as a Medical Device (SaMD). 2018. Available online: <https://www.fda.gov/medical-devices/digital-health-center-excellence/software-medical-device-samd> (accessed on 2 July 2021).
15. FDA. Proposed Regulatory Framework for Modifications to Artificial Intelligence/Machine Learning (AI/ML)-Based Software as a Medical Device (SaMD)—Discussion Paper and Request for Feedback; U.S Food and Drug Administration. 2019; pp. 1–20. Available online: <https://www.fda.gov/media/122535/download> (accessed on 2 July 2021).
16. Benjamens, S.; Dhunnoo, P.; Meskó, B. The state of artificial intelligence-based FDA-approved medical devices and algorithms: An online database. *NPJ Digit. Med.* **2020**, *3*, 118. [[CrossRef](#)]
17. *Standard IEC 62304:2006; Medical Device Software—Software Life Cycle Processes*. International Organization for Standardization: Geneva, Switzerland 2006.
18. *Standard ISO 14971:2019; Medical Devices—Application of Risk Management to Medical Devices*. International Organization for Standardization: Geneva, Switzerland 2019.
19. Rowley, A.; Turpin, R.; Walton, S. *The Emergence of Artificial Intelligence and Machine Learning Algorithms in Healthcare: Recommendations to Support Governance and Regulation*; Position Paper; BSI Group: London, UK 2019; pp. 1–18.
20. Baird, P.; Hoefler, E.; Lewelling, J.; Turpin, R. *Machine Learning AI in Medical Devices: Adapting Regulatory Frameworks and Standards to Ensure Safety and Performance*; White Paper; AAMI and BSI: Arlington, VA, USA 2020; pp. 1–24.
21. Cheung, M.; Campbell, J.J.; Thomas, R.J.; Braybrook, J.; Petzing, J. Systematic design, generation, and application of synthetic datasets for flow cytometry. *PDA J. Pharm. Sci. Technol.* **2022**. [[CrossRef](#)]
22. Qiu, W.; Joe, H. *clusterGeneration: Random Cluster Generation (with Specified Degree of Separation)*; R Package Version 1.3.5. 2020. Available online: <https://CRAN.R-project.org/package=clusterGeneration> (accessed on 2 July 2021).
23. Hahne, F.; LeMeur, N.; Brinkman, R.R.; Ellis, B.; Haaland, P.; Sarkar, D.; Spidlen, J.; Strain, E.; Gentleman, R. flowCore: A Bioconductor package for high throughput flow cytometry. *BMC Bioinform.* **2009**, *10*, 106. [[CrossRef](#)]
24. Azzalini, A. *sn: The Skew-Normal and Related Distributions such as the Skew-t*; R Package Version 1.6-2; Università di Padova: Padua, Italy, 2020.
25. Wickham, H.; Seidel, D. *scales: Scale Functions for Visualization*; R Package Version 1.1.1. 2020. Available online: <https://CRAN.R-project.org/package=scales> (accessed on 2 July 2021).

26. Revelle, W. *psych: Procedures for Psychological, Psychometric, and Personality Research*; R Package Version 2.0.12; Northwestern University: Evanston, IL, USA, 2020.
27. Qian, Y.; Wei, C.; Eun-Hyung Lee, F.; Campbell, J.; Halliley, J.; Lee, J.A.; Cai, J.; Kong, Y.M.; Sadat, E.; Thomson, E.; et al. Elucidation of seventeen human peripheral blood B-cell subsets and quantification of the tetanus response using a density-based method for the automated identification of cell populations in multidimensional flow cytometry data. *Cytom. Part B Clin. Cytom.* **2010**, *78B*, S69–S82. [[CrossRef](#)]
28. Bhattacharya, S.; Dunn, P.; Thomas, C.G.; Smith, B.; Schaefer, H.; Chen, J.; Hu, Z.; Zalocusky, K.A.; Shankar, R.D.; Shen-Orr, S.S.; et al. ImmPort, toward repurposing of open access immunological assay data for translational and clinical research. *Sci. Data* **2018**, *5*, 1–9. [[CrossRef](#)]
29. Aghaeepour, N.; Nikolic, R.; Hoos, H.H.; Brinkman, R.R. Rapid cell population identification in flow cytometry data. *Cytom. Part A* **2011**, *79A*, 6–13. [[CrossRef](#)]
30. Van Gassen, S.; Callebaut, B.; Van Helden, M.J.; Lambrecht, B.N.; Demeester, P.; Dhaene, T.; Saeys, Y. FlowSOM: Using self-organizing maps for visualization and interpretation of cytometry data. *Cytom. Part A* **2015**, *87*, 636–645. [[CrossRef](#)]
31. Levine, J.H.; Simonds, E.F.; Bendall, S.C.; Davis, K.L.; Amir, E.A.D.; Tadmor, M.D.; Litvin, O.; Fienberg, H.G.; Jager, A.; Zunder, E.R.; et al. Data-driven phenotypic dissection of AML reveals progenitor-like cells that correlate with prognosis. *Cell* **2015**, *162*, 184–197. [[CrossRef](#)]
32. Chen, H. Rphenograph: R Implementation of the Phenograph Algorithm, 2015. R Package Version 0.99.1. Available online: <https://github.com/JinmiaoChenLab/Rphenograph> (accessed on 2 July 2021).
33. Qiu, P.; Simonds, E.F.; Bendall, S.C.; Gibbs, K.D.; Bruggner, R.V.; Linderman, M.D.; Sachs, K.; Nolan, G.P.; Plevritis, S.K. Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. *Nat. Biotechnol.* **2011**, *29*, 886–893. [[CrossRef](#)]
34. Qiu, P. Toward deterministic and semiautomated SPADE analysis. *Cytom. Part A* **2017**, *91*, 281–289. cyto.a.23068. [[CrossRef](#)]
35. Naim, I.; Datta, S.; Rebhahn, J.; Cavanaugh, J.S.; Mosmann, T.R.; Sharma, G. SWIFT-scalable clustering for automated identification of rare cell populations in large, high-dimensional flow cytometry datasets, Part 1: Algorithm design. *Cytom. Part A* **2014**, *85*, 408–421. [[CrossRef](#)]
36. Mosmann, T.R.; Naim, I.; Rebhahn, J.; Datta, S.; Cavanaugh, J.S.; Weaver, J.M.; Sharma, G. SWIFT-scalable clustering for automated identification of rare cell populations in large, high-dimensional flow cytometry datasets, Part 2: Biological evaluation. *Cytom. Part A* **2014**, *85*, 422–433. [[CrossRef](#)] [[PubMed](#)]
37. Malek, M.; Taghiyar, M.J.; Chong, L.; Finak, G.; Gottardo, R.; Brinkman, R.R. flowDensity: Reproducing manual gating of flow cytometry data by automated density-based cell population identification. *Bioinformatics* **2015**, *31*, 606–607. [[CrossRef](#)] [[PubMed](#)]
38. Demaret, J.; Varlet, P.; Trauet, J.; Beauvais, D.; Grossemy, A.; Hégo, F.; Yakoub-Agha, I.; Labalette, M. Monitoring CAR T-cells using flow cytometry. *Cytom. Part B Clin. Cytom.* **2021**, *100*, 218–224. [[CrossRef](#)] [[PubMed](#)]
39. Lorimer, T.; Held, J.; Stoop, R. Clustering: How much bias do we need? *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2017**, *375*, 20160293. [[CrossRef](#)] [[PubMed](#)]